```
entry:
                                                                   call void @llvm.dbg.value(metadata ptr %func args, metadata !78, metadata
                                                                   ...!DIExpression()), !dbg !86
                                                                   %call = tail call i32 @initialise arrays(ptr noundef nonnull @ func .s253)
                                                                   ... #8, !dbg !87
                                                                   %call1 = tail call i32 @gettimeofday(ptr noundef %func args, ptr noundef
                                                                   ... null) #8, !dbg !88
                                                                   call void @llvm.dbg.value(metadata i32 0, metadata !80, metadata
                                                                   ... !DIExpression()), !dbg !89
                                                                   br label %for.cond2.preheader, !dbg !90
                                                                    for.cond2.preheader:
                                                                     %nl.037 = phi i32 [ 0, %entry ], [ %inc20, %for.cond.cleanup4 ]
                                                                     call void @llvm.dbg.value(metadata i32 %nl.037, metadata !80, metadata
                                                                     ...!DIExpression()),!dbq!89
                                                                     call void @llvm.dbg.value(metadata i32 0, metadata !82, metadata
                                                                     ... !DIExpression()), !dbg !91
                                                                     %0 = call i32 @llvm.vscale.i32(), !dbg !92
                                                                     %1 = \text{shl i} 32 \%0, 2, !\text{dbg !} 92
                                                                     %2 = icmp uge i32 %nl.037, %1, !dbg !92
                                                                     br i1 %2, label %Pre.Vectorization, label
                                                                     ... %Preheader.for.remaining.iterations, !dbg !92
                                                                                                                            F
                                  Pre. Vectorization:
                                  %8 = \text{call i32 @llvm.vscale.i32()}
                                  %9 = call <vscale x 4 x i32> @llvm.experimental.stepvector.nxv4i32()
                                  %step.value = shl i32 \%8, 2
                                  %10 = urem i32 %nl.037, %step.value
                                  %total.iterations.to.be.vectorized = sub i32 %nl.037, %10
                                  %11 = insertelement <vscale x 4 x i32> poison, i32 %step.value, i64 0
                                  %stepVector.update.values = shufflevector <vscale x 4 x i32> %11, <vscale x
                                  ... 4 x i32> poison, <vscale x 4 x i32> zeroinitializer
                                  br label %vectorizing.block
                               vectorizing.block:
                               %12 = phi i32 [ 0, %Pre.Vectorization ], [ %26, %vectorizing.block ]
                               %13 = phi < vscale \times 4 \times i32 > [\%9, \%Pre. Vectorization], [\%27,
                               ... %vectorizing.block ]
                               %14 = getelementptr inbounds [32000 x i32], ptr @a, i64 0, i32 %12, !dbg !103
                               %15 = getelementptr inbounds [32000 x i32], ptr @b, i64 0, i32 %12, !dbg !111
                               %16 = load < vscale x 4 x i32 >, ptr %14, align 16
                               %17 = load < vscale x 4 x i32 >, ptr %15, align 16
                               %19 = getelementptr inbounds [32000 x i32], ptr @d, i64 0, i32 %12, !dbg !114
                               %20 = getelementptr inbounds [32000 x i32], ptr @c, i64 0, i32 %12, !dbg !118
                               %21 = call <vscale x 4 x i32> @llvm.aarch64.sve.ld1.nxv4i32(<vscale x 4 x
                               ... i1> %18, ptr %19)
                               %22 = call <vscale x 4 x i32> @llvm.aarch64.sve.mul.nxv4i32(<vscale x 4 x
                               ... i1> %18, <vscale x 4 x i32> %21, <vscale x 4 x i32> %17)
                               %23 = call <vscale x 4 x i32> @llvm.aarch64.sve.sub.nxv4i32(<vscale x 4 x
                               ... i1> %18, <vscale x 4 x i32> %16, <vscale x 4 x i32> %22)
                               \%24 = \text{call} < \text{vscale x 4 x i} 32 > \text{@llvm.aarch} 64.\text{sve.ld1.nxv4} i 32 (< \text{vscale x 4 x})
                               ... i1> %18, ptr %20)
                               %25 = call <vscale x 4 x i32> @llvm.aarch64.sve.add.nxv4i32(<vscale x 4 x
                               ... i1> %18, <vscale x 4 x i32> %23, <vscale x 4 x i32> %24)
                               call void @llvm.aarch64.sve.st1.nxv4i32(<vscale x 4 x i32> %25, <vscale x 4
                               ... x i1> %18, ptr %20)
                               call void @llvm.aarch64.sve.st1.nxv4i32(<vscale x 4 x i32> %23, <vscale x 4
                               ... x i1> %18, ptr %14)
                               %26 = add i32 %step.value, %12
                               %27 = add <vscale x 4 x i32> %13, %stepVector.update.values
                               %terminate.condition = icmp uge i32 %26, %total.iterations.to.be.vectorized
                               br i1 %terminate.condition, label %middle.block, label %vectorizing.block
                          middle.block:
                          %condition = icmp eq i32 %10, 0
                          \%7 = \text{zext i} 32 \% 26 \text{ to i} 64
                          br i1 %condition, label %for.cond.cleanup4, label
                          ... %Preheader.for.remaining.iterations
                                     Т
                                           Preheader.for.remaining.iterations:
                                           %28 = phi i64 [ 0, %for.cond2.preheader ], [ %7, %middle.block ]
                                           br label %for.body5
                                     for.body5:
                                     %indvars.iv = phi i64 [ %indvars.iv.next, %for.inc ], [ %28,
                                     ... %Preheader.for.remaining.iterations ]
                                     call void @llvm.dbg.value(metadata i64 %indvars.iv, metadata !82, metadata
                                     ...!DIExpression()), !dbg !91
                                     %arrayidx = getelementptr inbounds [32000 x i32], ptr @a, i64 0, i64
                                     ... %indvars.iv, !dbg !103
                                     %3 = load i32, ptr %arrayidx, align 4, !dbg !103, !tbaa !107
                                     %arrayidx7 = getelementptr inbounds [32000 x i32], ptr @b, i64 0, i64
                                     ... %indvars.iv, !dbg !111
                                     %4 = load i32, ptr %arrayidx7, align 4, !dbg !111, !tbaa !107
                                     %cmp8 = icmp sgt i32 %3, %4, !dbg !112
                                     br i1 %cmp8, label %if.then, label %for.inc, !dbg !113
                                                                                               F
                  if.then:
                   %arrayidx14 = getelementptr inbounds [32000 x i32], ptr @d, i64 0, i64
                   ... %indvars.iv, !dbg !114
                   %5 = load i32, ptr %arrayidx14, align 4, !dbg !114, !tbaa !107
                   %mul = mul nsw i32 %5, %4, !dbg !116
                   %sub = sub nsw i32 %3, %mul, !dbg !117
                   call void @llvm.dbg.value(metadata i32 %sub, metadata !79, metadata
                   ...!DIExpression()),!dbg!86
                   %arrayidx16 = getelementptr inbounds [32000 x i32], ptr @c, i64 0, i64 ... %indvars.iv, !dbg !118
                   %6 = load i32, ptr %arrayidx16, align 4, !dbg !119, !tbaa !107
                   %add = add nsw i32 %sub, %6, !dbg !119
                   store i32 %add, ptr %arrayidx16, align 4, !dbg !119, !tbaa !107
                   store i32 %sub, ptr %arrayidx, align 4, !dbg !120, !tbaa !107
                   br label %for.inc, !dbg !121
                                        for.inc:
                                        %indvars.iv.next = add nuw nsw i64 %indvars.iv, 1, !dbg !122
                                        call void @llvm.dbg.value(metadata i64 %indvars.iv.next, metadata !82,
                                        .. metadata !DIExpression()), !dbg !91
                                        %exitcond.not = icmp eq i64 %indvars.iv.next, 32000, !dbg !123
                                        br i1 %exitcond.not, label %for.cond.cleanup4, label %for.body5, !dbg !92,
                                        ...!llvm.loop!124
                   for.cond.cleanup4:
                    %inc20 = add nuw nsw i32 %nl.037, 1, !dbg !97
                    call void @llvm.dbg.value(metadata i32 %inc20, metadata !80, metadata
                    ..!DIExpression()),!dbq!89
                    %exitcond39.not = icmp eq i32 %inc20, 100000, !dbg !98
                    br i1 %exitcond39.not, label %for.cond.cleanup, label %for.cond2.preheader,
                    ...!dbg!90,!llvm.loop!99
                                     Τ
%t2 = getelementptr inbounds %struct.args t, ptr %func args, i64 0, i32 1,
%call22 = tail call i32 @gettimeofday(ptr noundef nonnull %t2, ptr noundef
%call23 = tail call i32 @calc checksum(ptr noundef nonnull @ func .s253)
```

CFG for 's253' function

for.cond.cleanup:

... #8, !dbg !95

... null) #8, !dbg !94

ret i32 %call23, !dbg !96

... !dba !93